THESIS FOR THE DEGREE OF LICENTIATE OF ENGINEERING

Interactive Optimisation in Marine Propeller Design

IOLI GYPA

Department of Mechanics and Maritime Sciences
Division of Marine Technology
CHALMERS UNIVERSITY OF TECHNOLOGY

Göteborg, Sweden 2021

Interactive Optimisation in Marine Propeller Design
IOLI GYPA

Thesis for the degree of Licentiate of Engineering 2021:12
ISSN 1652-8565
Department of Mechanics and Maritime Sciences
Division of Marine Technology
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone: +46 (0)31-772 1000

Chalmers Reproservice
Göteborg, Sweden 2021

Interactive Optimisation in Marine Propeller Design
IOLI GYPA
Department of Mechanics and Maritime Sciences
Division of Marine Technology
Chalmers University of Technology

Abstract

Marine propeller design is a complex engineering problem that depends on the collaboration of several scientific disciplines. During the design process, the blade designers need to consider contradicting requirements and come up with one optimal propeller design as a solution to the specific problem. This solution is usually the trade-off between the stakeholders' requirements and the objectives and constraints of the problem.

The significant amount of design variables related to blade design problems requires a systematic search in a large design space. Automated optimisation has been utilised for a number of blade design applications, as it has the advantage of creating a large set of design alternatives in a short period of time. However, automated optimisation has failed to be used in industrial applications, due to its complex set-up and the fact that in more complex scenarios the majority of the non-dominated design alternatives are infeasible. This necessitates a way of enabling the blade designers to interact with the algorithm during the optimisation process.

The purpose of this thesis is to develop a methodology that supports the blade designers during the design process and to enable them to interact with the design tools and assess design characteristics during the optimisation. The overall aim is to improve the design performance and speed. According to the proposed methodology, blade designers are called during intermediate stages of the optimisation to provide information about the designs and then this information is input in the algorithm. The goal is to steer the optimisation to an area of the design space with feasible Pareto designs, based on the designer's preference. Since there are objectives and constraints that cannot be quantified with the available computational tools, keeping the "human in the loop" is essential, as a means to obtain feasible designs and quickly eliminate designs that are impractical or unrealistic.

The results of this research suggest that through the proposed methodology the designers have more control over the whole optimisation procedure and they obtain detailed Pareto frontiers that involve designs that are characterised by high performance and follow the user preference.

Keywords: marine propeller design, optimisation, NSGA-II, progressively interactive evolutionary computation, interactive genetic algorithms, machine learning, support-vector machines

Preface

This thesis is a summary of the research work carried out at the Division of Marine Technology, Department of Mechanics and Maritime Sciences at the Chalmers University of Technology during the years 2018-2021. Funding for this research was provided by Vinnova through UnikAI, by Chalmers University of Technology Foundation for the strategic research project Hydro- and aerodynamics; by the Swedish Transportation Agency via Lighthouse through the SailProp project; and by Kongsberg Maritime Sweden AB through the University Technology Centre in Computational Hydrodynamics hosted at the Department of Mechanics and Maritime Sciences at Chalmers.

I would like to sincerely thank my supervisor Professor Rickard Bensow for his constant guidance and assistance during my research project and for always finding time to support me, especially at the times that the research did not go as planned. I would also like to thank my co-supervisor Associate Professor Krister Wolff for the interesting discussions on optimisation. I am very grateful for the support I have received from several people at Kongsberg Maritime Sweden AB, especially from Marcus Jansson for his huge help and scientific contribution to my project. I would also like to thank Göran Grunditz, Johan Lundberg and Robert Gustafsson for sharing their blade design knowledge and experience with me.

I would like to thank my colleagues from the division of Marine Technology for creating a nice and inspiring working environment. Special thanks to Francesco, Mohammad and Mohsen for being such supportive friends.

Finally, I would like to thank my supportive and lovely parents, my brother Foivos who is my role model and my awesome friends. Anastasia and Leo, I cannot thank you enough for always being there for me.

Gothenburg, June 2021
Ioli Gypa

List of appended papers

This thesis consists of an extended summary and the following appended papers:

Paper A	I. Gypa, R. Bensow, K. Wolff, and R. Gustafsson. "Interactive Evolutionary Computation for Propeller Design Optimization of Wind-Assisted Vessels". *AIAA AVIATION 2020 FORUM*. 2020. DOI: 10.2514/6.2020-3162
Paper B	I. Gypa, M. Jansson, K. Wolff, and R. Bensow. "Propeller optimisation by interactive genetic algorithms and machine learning". Manuscript submitted for publication in Ship Technology Research journal.

Contents

Part I
Extended Summary

1 Introduction

Waterborne transport is responsible for 90% of the world's trade nowadays and therefore, designing ships that travel safely and efficiently should be the responsibility and goal of the shipping industry on a global scale. The environmental impact of shipping is huge though, and in recent years, it has been urgent to add another goal: designing vessels in a sustainable manner with the aim to achieve the decarbonisation of the industry. Focusing on the development of more efficient technologies in all stages of the ship design process, contributes towards attaining this goal.

The selection of the propulsion system is an important part of the ship design process and the overall goal is to design a unique, efficient propeller, which matches the hull and machinery system, creates the required thrust and fulfils the requirements, set from the stakeholders. The requirements from each stakeholder are different and often contradicting and the blade designers need to take them into consideration and decide on a geometry that is the best compromise between all these requirements. Except propeller efficiency, other crucial requirements are fuel consumption, overall cost, comfort, cavitation, propeller-hull induced pressure pulses, classification regulations etc. All things considered, the blade design process in an industrial framework has to be straightforward, well-developed and be completed in a limited time frame.

1.1 Motivation and Objectives

Engineering systems have increasingly become more difficult to design and manage, due to their complexity. They combine several disciplines that require collaboration, which is a difficult task to achieve successfully, since a large number of design variables, complex physics phenomena, computationally expensive simulations etc. need to be handled in parallel. Multidisciplinary design optimisation (MDO) processes are used as a tool for solving problems of complex engineering systems. The goals are set as objective functions in a systematic way and through optimisation, the best solutions are found. However, objective functions do not always achieve solutions that satisfy all the components of the systems. The importance of involving the human in MDO processes has increased nowadays, due to the human ability of eliminating ineffective solutions and understanding the aim of each component [26].

Blade design is a MDO process that combines many disciplines and in parallel has several objectives, related to the previously mentioned requirements. Its wide design space and contradicting requirements have made the blade design process very complex to handle and automated optimisation has proved to be insufficient for complex scenarios in industrial applications. Including blade designers in the optimisation process appears to be more necessary.

The purpose of the thesis is to develop a methodology that supports the blade designers during the design process and enables them to interact with the design tools and assess design characteristics during the optimisation, with the aim to improve the design performance and speed.

With the proposed methodology, the behaviour of a user-guided interactive optimisation method is investigated as one component in an improved industrial propeller design process. Objectives of the proposed methodology are the following:

- Assess cavitation characteristics by displaying the cavity shape on the blade to the blade designers instead of using quantitative cavitation constraints.
- Implement the user-code interaction in the existing optimisation methodology by using interactive evolutionary computation.
- Solve the problem of user-fatigue that is connected to interactive optimisation.
- Obtain a detailed Pareto frontier at the end of the optimisation with efficient and realistic propeller geometries in a short time frame.

This process aims at empowering the blade designers, instead on substituting them, and gives them the opportunity to have control over the whole optimisation process.

1.2 Outline

This thesis is a summary of the appended papers A and B. Paper A is the first step of the proposed methodology towards interactive optimisation for a blade design problem. The designers select interesting areas in the Pareto plots and prioritise them for the following generations. In paper B, the methodology is extended and involves one more interactive step, where the designers assess cavitation characteristics. A support-vector machine model is also implemented as surrogate model in the methodology. The results of the two appended papers are reported in chapter 5.

The rest of the thesis is organised as follows. In chapter 2, the different stages of the marine propeller design process are described, with focus on an industrial design task. A review on automated optimisation for several blade design applications is done and the benefits and limitations of automated approaches are discussed. Chapter 3 presents the interactive optimisation and the various approaches of the interactive evolutionary computation with a literature review on several applications. The proposed methodology is explained step by step in chapter 4 and the mathematical model of the support-vector machines is briefly described. Finally, there is a discussion on the conclusions for the proposed methodology in chapter 6 and on how it can be further developed.

2 Marine propeller design process

2.1 Industrial design task and current procedures

The selection and design of a marine propeller is a complex procedure that involves several stakeholders and requires expertise in different scientific fields. Due to its multidisciplinary and multi-objective nature, the final blade design depicts a trade-off for often contradicting techno-economic requirements, objectives and constraints. Propeller design is an important part of the ship design spiral, and its process can be represented from a spiral as well, since it is iterative and entails several stages that interact and interconnect. The design process can be divided in the three following main stages, which will be described in detail:

- Concept design
- Preliminary design
- Detailed design

After the completion of the design process, the blade designers decide on one propeller that will later be manufactured.

2.1.1 Concept Design

The propeller selection and design process start with the concept design stage. The customer (shipping company, ship owner, shipyard etc.) informs the propeller supplier on the vessel's mission along with its propulsion needs. The aim of this stage is to translate the mission requirements into realistic propulsion characteristics, in order to select the correct propeller type and design point. The customer will set requirements that are usually related to efficiency, fuel consumption, costs, comfort, etc. and subsequently the blade designers will consider additional requirements, such us cavitation, propeller induced pressure pulses, classification regulations etc. Necessary input here is the vessel type along with its main dimensions and hull characteristics, the mission profile that includes operating conditions, ship route and service life and the requirements related to the engine and machinery system. Information related to ship resistance along with data from model tests are essential and they usually have been defined or collected in earlier stages of the ship design spiral. However, it is possible that some important data are lacking, hence empirical formulas can be used or CFD simulations might be performed, in order to extract the missing information.

Matching the propeller towards the hull and the machinery system is essential in order to fulfil the power requirements and attain the desirable performance. The majority of the conventional cargo ships operate for the most part of their voyages under one condition (design condition) with a specific speed. However, for the design of the propeller, all operational conditions need to be taken into consideration and the designer has to select a suitable design point that will lead to good performance of the engine even for off-design conditions that the vessel will encounter during its service life. Hence, a sea margin of 10-25% is applied, in order to take into account conditions with increased resistance due to the vessel's loading, harsh sea state, hull and propeller roughness, shallow waters, trimming etc.[1]; this leads to having more than one propeller demand curve. In addition to this, an engine margin of 10-15% of the maximum continuous rating (MCR) is applied, in order to decrease fuel costs and enable increased power for off-design conditions [1]. Different blade designers might select different design points for the same problem, something that will eventually lead to a different propeller design. Therefore, the selection of the right design point requires great attention. For commercial propeller suppliers that have large databases with designs from older projects, it is common practice to use this information for the selection of the design point for projects/vessels with similar geometry and mission requirements. Information from older propeller designs is also used in the next design stages, especially in cases where not all essential data are available.

2.1.2 Preliminary Design

Once the propeller type and design point have been selected, the next stage is the preliminary design of the propeller, where the aim is to define the main propeller particulars: the propeller diameter, number of blades, mean pitch, blade area ratio and the sectional ratios of pitch, camber, thickness, skew, rake and chord length. The selection of the main particulars is directly linked to the requirements set on the previous design stage. For example, if high efficiency is the goal, then the blade designers would choose a large propeller diameter with less blades and decreased blade area ratio. In yacht design, where usually comfort is the goal, meaning less noise and vibrations, larger propeller – hull clearance is preferred in order to achieve lower hull pressure pulses, in combination with a higher number of blades. If low risk of cavitation erosion is a requirement, blades with higher blade area ratio would be preferred in this design stage [30]. Note that there is more focus on cavitation during the next design phase, as the more detailed design parameters have a greater effect on cavitation.

Circulation theory and the use of systematic propeller series are the tools that aid the designers during this stage to choose the main particulars [34]. The designers choose some initial values for the blade area ratio, thickness at the midchord and the tip, as well as the skew and the rake distributions. It is common practice at this point to reduce the tip loading of the blade, in order to obtain reduced pressure pulses later. Then, the optimal pitch and camber distributions are calculated with the aid of lifting line and lifting surface methods respectively. The aforementioned distributions are represented by spline curves along the radius of the propeller and the designers should always check how the curvature of the splines is formed. A means to check if the designer follows the right direction in selecting the main particulars is to verify that the midchord bubble cavitation is within the required limits that have been set out of experience. Consequently, the whole process is iterated until the cavitation requirements are met. If this is not possible, then the designers return to the concept design stage, redefine the design point and then restart the preliminary design process with the new input.

2.1.3 Detailed Design

The purpose of this stage is to determine the final detailed geometry of the propeller that will later be manufactured, along with information about the hub. Selecting correctly all the detailed design parameters will lead to successfully fulfilling the requirements that were set during concept design. The final outcome is a unique propeller for the specific vessel that achieves all objectives as effectively as possible.

The output parameters of the preliminary design, together with the information about the wake from the model tests or the simulations, constitute the input of the detailed design stage. As a first step, the designers need to select the suitable design parameters in order to achieve a fitting position of the blade on the flange of the hub and to avoid blade collision. In addition to this, visualising these geometrical characteristics in plots is beneficial and it speeds up the design process. Small alterations in the design parameters are done iteratively until the correct position is achieved. The next step involves the calculation of the static and dynamic strength of the blades. This can be accomplished by utilising different numerical methods, such as beam theory, or finite element method (FEM) tools. Moreover, the blade's thickness is defined by following the rules of classification societies. Finally, the designers need to analyse the hydrodynamic performance of the propeller in order to calculate the propeller forces and as a result the power consumption. This analysis can be done through different types of numerical methods, but at this stage potential methods are usually preferred, as they are very fast. Except the efficiency prediction, an important part of this analysis is the evaluation of sheet cavitation, since it can lead to potential erosion damages. The designers visualise graphically how the cavitation has been developed on the blade and assess whether it is within the satisfactory limits or not. Additionally, according to the mission requirements, the propeller-induced pressure pulses can be calculated either through empirical methods like Holden [14] or through the above-mentioned numerical methods. If one of these requirements/objectives are not fulfilled or if the designers are not fully satisfied with the results they receive, then they iterate the procedure of the detailed design until the desirable results are obtained. If this is not possible, then the designers need to return to the preliminary design or in some cases even to the concept design and redefine the important design parameters.

During the design process, there are parts, where it is possible to change geometrical parameters systematically until the designers manage to attain specific objectives. At the same time, there are other parts that may not vary systematically, and the designers need to visualise and evaluate some characteristics of the propeller. The blade designers have the control of the whole procedure when they follow a manual blade design process and this is the most common practice for the industry nowadays. However, creating several design alternatives can become very labour-intensive. Automating the whole design process, with the aim to create many designs, has been used as a solution to this problem, mostly in academic applications. This is commonly done with the aid of various optimisation algorithms. The result of the optimisation is a set of optimal designs, also known as Pareto frontier. The designers decide which designs offer the best trade-off according to the requirements, objectives and constraints that were set in earlier design stages. In many cases, depending on the project, the designers will evaluate the designs further with tools using numerical methods of higher fidelity and finally they will decide on the best design for the specific ship and according to the requirements.

2.2 Automated Optimisation

Automated optimisation is utilised in several different types of complex engineering systems, as a tool to aid the engineers in finding the optimal solutions for the problems they encounter in limited time. As previously described, blade design is dependent on various geometrical parameters, it is an iterative process and requires the control of the output parameters in between the different design stages. Another characteristic, especially when blade design is performed in an industrial framework, is that the process runs under strict time constraints. Manual blade design has proved to be a labour-intensive process, due to its contradictive requirements that the designers need to consider. Therefore, automated optimisation has been utilised in recent years in a number of applications, as a means to enable blade designers to obtain a large number of design alternatives in a short period of time.

The general concept of automated optimisation is that the most decisive design characteristics, in terms of system performance, are being parameterised. By changing the values of the design parameters, new designs/solutions are created, while other parameters remain constant. This can be done more systematically with the use of optimisation algorithms in order to search the design space efficiently and find one or a set of optimal designs. Although deterministic optimisation algorithms search the whole design space and would potentially find the optimal solution to the problem, it is in practice impossible to use them in most engineering problems. Population-based, nature-inspired stochastic optimisation algorithms, like genetic algorithms (GAs), particle swarm optimisation (PSO), ant colony optimisation (ACO), are primarily utilised, since they cleverly guide the exploration to areas of the design space with the optimal solutions in a fast manner. The optimisation eventually provides the designers with a set of optimal designs, which are referred to as Pareto frontier. All the solutions of the Pareto frontier are non-dominated, and the selection of a unique solution/design depends on the decision of the designers. Different designers might select a different solution according to their preference and experience.

Many different implementations of automated optimisation in blade design have been developed both in academia and in the industry. A common feature is that although complex phenomena are involved, the analysis tools should be fast, meaning that high-fidelity simulations or experiments are not feasible. Thus, numerical methods of lower fidelity are the tools that are primarily used here, so that the optimisation process is quick. Different combinations of numerical methods with optimisation algorithms have been implemented, depending on the requirements of the problem. Berger et al. [2] developed a two stage optimisation methodology, where in the first stage an evolutionary algorithm was used for the optimisation and an in-house panel method was used for the hydrodynamic analysis of the designs. Some of the optimal designs were selected and were further analysed during the second stage by a numerical method that combined a viscous flow solver with a boundary element method (BEM). Vesting et al. [35, 36] developed a fully automated optimisation process that involved constraint handling using semi-empirical analysis of cavitation nuisance. A Non-dominated Sorting Genetic Algorithm II (NSGA-II) and a PSO algorithm were utilised together with a vortex lattice method (VLM) tool for the calculation of the propeller's hydrodynamic performance. Huisman and Foeth [15] used the NSGA-II algorithm for the optimisation together with a BEM for two different propeller optimisation problems. Gaggero et al. [9] combined a GA with a BEM tool and a viscous flow solver for the multi-objective design optimisation of a propeller for a high-speed craft. A more practical optimisation tool was developed by Valdenazzi et al. [31], for the hydro-acoustic optimisation of naval propellers. The optimisation was performed in three levels, where the multi-objective GA was utilised, coupled with a BEM tool in levels 1 and 2 and a CFD code of higher fidelity in level 3.

The primary advantage in automated optimisation is that the optimisation algorithms efficiently search the most beneficial areas of the design space. After defining the most important design variables that control the optimisation, the process runs, and the designers obtain the Pareto frontier as output. In the meantime, they are not involved with the code, while deep knowledge on the optimisation process is not necessary. However, automated optimisation has failed to be utilised in an industrial framework. The definition of the design space can get complex and hard to set-up for different propeller designs. Design of experiments are usually utilised for defining the most important design variables and areas of the design space, but their execution takes considerable time. In addition to this, the majority of the Pareto frontier solutions are infeasible, since using potential numerical methods combined with semi-empirical analysis of cavitation, often can lead to inefficient propeller designs. Moreover, in complex problems that entail, for example, many objectives and constraints for several operating conditions, it is difficult to develop a single optimisation case to solve them [17]. All this has resulted in considering the traditional manual design process as more reliable and easier to handle, compared to the automated optimisation [35].

Most of the above-mentioned applications have found an intermediate step necessary, which involves the blade designers in different ways in the blade design optimisation process. However, it has not been integrated in the optimisation in a systematic and natural way. A way to achieving this goal, is through different methods of the interactive evolutionary computation [29], which is described in detail in chapter 3.

3 Interactive Optimisation

The more complex the engineering systems get, the need for involving the designers as a part of those systems grows [26]. Multidisciplinary and multi-objective processes can be simplified by integrating the human thought and knowledge in the optimisation loop of the systems and in parallel find solutions more efficiently. Interactive evolutionary computation (IEC) is a method that enables the integration of the code - user interaction.

3.1 Description of IEC

IEC is an optimisation method that is based on evolutionary computation and utilises subjective human evaluation in its process. It is primarily implemented in systems that involve criteria that make the human assessment and analysis necessary. During the conventional evolutionary computation, the performance of the objectives is calculated through fitness functions. However, in IEC it is impossible to develop explicit fitness functions for those objectives that entail subjective criteria; thus the system users, who will be referred to as users, interact with some features of the system, judge them and return their input into the system. In this way, the new input is taken into account and affects the output of the system. The user interaction has several different types, such as figure visualisation, audio listening etc. A straightforward interface for user - machine interaction is a prerequisite in most systems, such that the interaction process is executed efficiently.

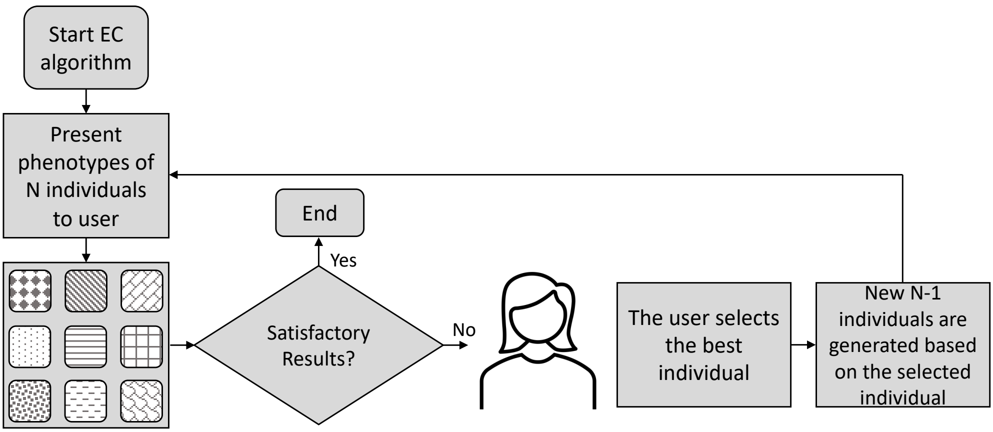


Figure 3.1: *Flowchart of a standard IEC Algorithm*

A flowchart of a standard IEC algorithm is presented in figure 3.1. As it is shown, the users start the optimisation with the aid of an EC algorithm and they pause it after some generations, and the phenotypes or output systems of N individuals are presented to the users. The phenotypes and the way they are presented to the users differ for each problem. Phenotypes can be simple patterns, symbols, artworks, sounds, music composition, physical representation of complex phenomena, etc. In the specific flowchart, images of different patterns are plotted on the screen. If the users are not fully satisfied with what they see, they should select the plot that is more suitable according to the problem. This information is input in the optimisation algorithm and the remaining N-1 individuals are discarded. The optimisation algorithm continues and generates new N-1 individuals, based on the selected individual. The phenotypes of the N-1 new individuals are plotted on the screen, together with the figure of the previously selected individual, which is placed in the middle of the screen. The process is iterated until the users obtain satisfactory results.

3.2 Background in IEC

IEC has been used over the years in several disciplines, like design, music, graphic arts, virtual reality, image processing, data mining and others [29]. In recent years, it has started being utilised for engineering, but mostly in applications where there are objectives related to engineering aesthetics, for example in car design [38].

Similarly with the automated optimisation, different types of EC algorithms can be utilised also here. Interactive genetic algorithms (IGAs) that are based on GAs are broadly used in various types of applications. They have been utilised for example for controller parameter optimisation in [8] with positive results in terms of performance and efficiency. The objective of the study was to tune the parameters of a proportional–integral–derivative regulator operating on a laboratory device, and in order to do that successfully the expertise of human operators was necessary. IGAs have also been used for product design, as presented in [18]. An interactive creative system for the conceptual design of a mobile phone was developed, with the aim to improve the design procedure at an early stage. Moreover, interactive PSO algorithms are used in several optimisation problems. In [19] an optimisation tool was presented, based on interactive PSO for the design of an airfoil and for the shape design of a compressor blade with several design parameters and multiple objectives. The focus of the study was on various types of visualisation techniques for the design space and the design features; for example heat maps and parallel coordinate visualisation plots were used to aid the code - user interaction process. In [7], interactive PSO was used for the architectural design of truss structures. The designers started with an initial design after setting up the optimisation, the PSO created more individuals that were presented to the users together with the structural weight and were compared to the initial design. The users selected the optimal designs iteratively until there was convergence in the solutions. Through the interactive PSO, the designers guided indeed the optimisation to an area with designs of lower structural weight and they matched the designer preference, as externally they were similar with the initial truss structure. Finally, interactive ACO has also been utilised in some applications, for example in [25] for early lifecycle software design, but according to the results it required more user interactions, compared to other IEC algorithms.

Except selecting the suitable IEC algorithm for an optimisation problem, a key parameter for setting up the IEC optimisation is the population size of the individuals.

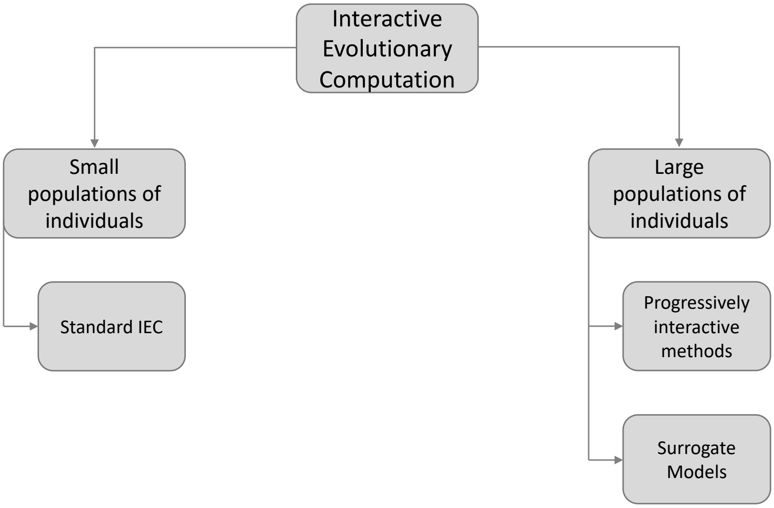


Figure 3.2: *Different categories of IEC methods*

The population size varies for different problems and applications. As it is shown in figure 3.2, when a small population size is utilised, a standard IEC approach is usually utilised, which is the method described in figure 3.1. When following a standard IEC approach, the whole population is presented to the users. For example, in [38] the users visualised all 100 individuals of each generation and ranked them with values 0-6. In [23], IEC was used in two modes, the automatic and the manual. During manual mode, the users evaluated the whole population, which consisted of 9 individuals for each one of the 30 generations. Also in [21], IEC was utilised for two process engineering problems and the users assessed all individuals of the population, where maximum seven individuals were presented at a time.

By evaluating the whole population of an optimisation problem, a fitness value is manually assigned to each individual, and the algorithm converges fast to a set of optimal solutions. However, there are several applications that require a search in a large design space and as a consequence a large population is needed. Blade design optimisation is an example of such applications that require large populations of blade designs. In those cases, the users do numerous manual evaluations, and after a point of visualising and assessing designs via a graphical user interface, human fatigue is caused and the users cannot evaluate the designs objectively any more. Human fatigue is the main disadvantage of interactive optimisation [37]. A problem that is directly linked to it is that only a subset of the solutions is presented to the users and some important solutions are often overlooked, which leads to increased convergence time of the algorithm [29]. There are two common approaches to solve those problems, as indicated in figure 3.2, the progressively interactive optimisation methods and the use of surrogate models.

For the progressively interactive optimisation, during the problem definition, an ideal point is defined by the users or by a function, as the reference point of the whole

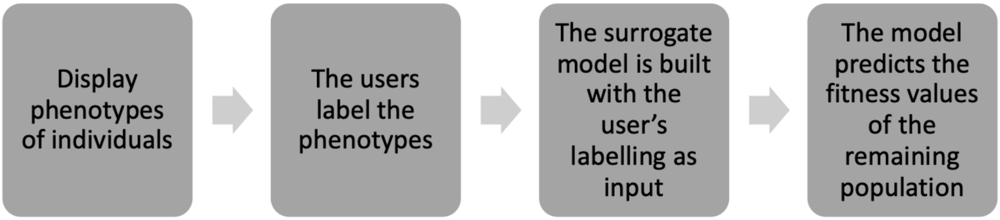


Figure 3.3: *Structure of surrogate model*

optimisation. Initially there is a large design space and as the optimisation evolves, the users are called in intermediate stages to guide the algorithm towards the reference solution. Every time the users provide new information, the design space is updated and narrowed down. Such an approach is implemented in [27], where the purpose of the study was to mitigate the number of user evaluations. The optimisation started with a large population and explored the design space; subsequently the users were called a number of times, the performance of the designs was presented to them and they provided information on the area of interest. After this point, the algorithm was switched to local search for exploitation of the subspace. The same approach was followed in [10], but the study was also focused on how to shape the areas of interest and which solutions of these areas to choose.

Progressively interactive methods are mainly used with the aim to narrow down the design space or focus in a specific area of the Pareto frontier. However, in many optimisation problems, part of the optimal solutions are inefficient and the user interaction is needed in order to visualise features of the solutions and distinguish the different solutions in a more realistic manner. In order to solve this problem and the user fatigue, IEC can be combined with surrogate models [28]. The users train the surrogate model by evaluating a subset of the entire dataset and the model approximates the performance of the remaining non-evaluated datapoints. The structure of a surrogate model is shown in figure 3.3. In this way, a system is created where large populations are produced through the EC algorithm, design features are presented to the users with the aid of the interactive optimisation and the surrogate model reduces the user fatigue.

There are several types of surrogate models used in IEC optimisation processes. In [28], a model was implemented that used two radial-basis function neural networks (NNs) and it was trained by a semi-supervised learning method. In [22], two methods were developed for predicting the performance of the non-evaluated solutions, a NN and Euclidean distances. In [3], IEC was used for a music application, where a surrogate model was implemented, since the users had to evaluate each audio separately, without being able to simultaneously make comparisons with other audios. The surrogate model was a NN with a cascade correlation technique. Support-vector machines (SVMs) have also been used as surrogate models for problems that entail classification. In [24], SVMs were implemented together with an efficient global optimisation algorithm and in [20] they were combined with partial ordering concepts and non-domination from multi-objective optimisation.

Deciding on whether the population size is small or large, depends a lot on how the user interface of the application has been developed, i.e. how many designs are being presented simultaneously, if there is a reference design for comparison, if the users are able to alternate the geometry of the designs etc. Focusing on the user interface of systems that use IEC methods, can lead to faster convergence of the algorithms and to more efficient solutions.

4 Interactive genetic algorithms supported by support-vector machines for propeller design optimisation

The propeller design process is multidisciplinary and multi-objective. Therefore, several design parameters are involved in the blade design optimisation process, which lead to the search of solutions in a wide design space. In order to search this space, large populations of blade design individuals need to be created by the optimisation algorithm. At the same time, the blade designers are being involved in the optimisation loop, with the aim to guide the algorithm with their input towards areas of the design space, where there are non-dominated solutions that represent realistic propeller designs.

In the methodology we have developed, we combine an IGA with a SVM model, used as surrogate model. The methodology follows the concept of progressively interactive optimisation, in the sense that the user-code interactions take place gradually (stepped optimisation). This is shown in figure 4.1, where the optimisation starts with a large population and the users are called in intermediate stages to evaluate the designs and with their selections, direct the optimisation to a detailed and well-defined Pareto frontier. In sections 4.1 and 4.2, the IGA methodology is described in detail, along with the mathematical model of the SVM model and its function in the IGA methodology.

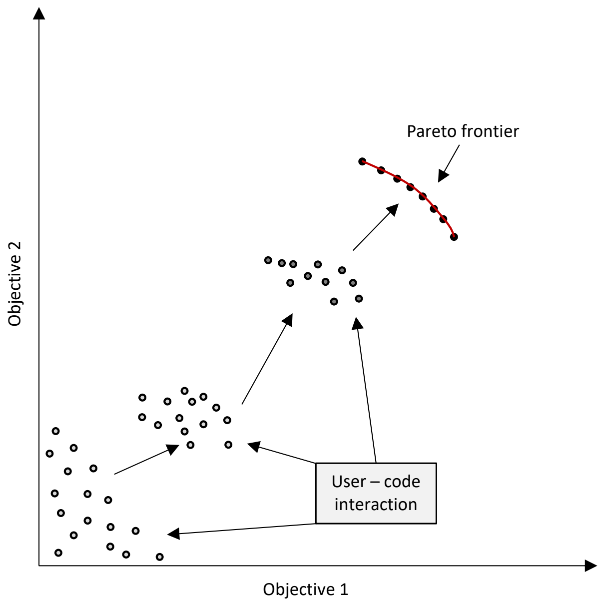


Figure 4.1: *Stepped IEC optimisation*

4.1 Methodology - Interactive Genetic Algorithms

The central concept of the methodology is that the blade designer manually creates a good design, which is referred to as baseline design, and begins an optimisation procedure, by setting the baseline design as the starting point. The overall aim is to eventually gain a set of improved designs, compared to the baseline. The user of the optimisation decides which should be the design variables for the specific problem and defines the objectives and constraints as well. The optimisation starts by using a GA and runs some time for a predefined number of generations, in order to produce a large amount of designs, before being paused. At this point, the interactive assessment of the designs by the user is essential. As previously described, there are some important design features that cannot be expressed accurately quantitatively, when using fast computational tools. Thus, through the proposed methodology, images of the design features are presented to the user and according to their experience and preference they accept or reject them. This new information is input in the GA and the optimisation continues, by prioritising the accepted design to continue to the following generations. The user can be called for assessment as often as it appears necessary according to the problem and the process finishes when the algorithm has reached good convergence with satisfactory performance, in terms of fitness and user preference. The SVM model is used for problems with very large amounts of designs, in order to accelerate the evaluation process. In short, the central concept is that large populations of individuals are created by the GA, user evaluations are enabled by the IEC in order to guide the optimisation towards areas of the design space with approved designs, and the user fatigue, caused by several manual assessments, is solved by using the SVM model. The flowchart of the methodology is shown in figure 4.2 and is described in detail below.

The optimisation starts by running the NSGA-II and after N predefined generations there is the evaluation stage, where images that present a design feature of the designs, are presented on the screen and the user decides whether they have an acceptable performance, according to this design feature. If the population of the designs is small, the plots of the features of all designs are shown to the user. Alternatively, since a large population would lead to numerous user evaluations, the SVM model is implemented, so that only a subset of the total number of solutions is used for user evaluations. The user assesses this subset and in this way the SVM model is trained and does a prediction for the remaining solutions. As a result, the evaluation process is accelerated. At the end of the evaluation stage, the non-satisfactory designs are rejected for both small and large populations of individuals.

Thereafter, the user has the option to update the whole optimisation set-up, which consists of the design variables, number of generations, individuals per generation, objectives, constraints and optimisation parameters. It is important to have the option to control some important parameters during the actual optimisation, instead of preconfiguring everything, as convergence can be achieved sooner. The number of individuals per generation is an important parameter, since the first generation of the following optimisation round consists of the accepted designs from the user evaluations or from the SVM model. As it is not known in advance how many designs will be accepted, changing the value of the population size is beneficial. If the user selects the following optimisation round to have a population size that is larger than the accepted designs, the missing individuals are created by using mutation and crossover on the accepted individuals. If they select a smaller population size, the accepted designs are prioritised by ranking them in the same way as in the NSGA-II. Finally, if they select the same number, all accepted designs create the first generation of the following round.

After updating the optimisation set-up, the optimisation resumes and it is paused after M generations, in order to present the performance of the Pareto designs together with their feature plots to the user, who in turn will decide whether the results are satisfactory or if the optimisation should resume by iterating the evaluation stage. The evaluation stage occurs as many times as it is needed in order to obtain satisfactory results or according to the available time and the fatigue from the user evaluations.

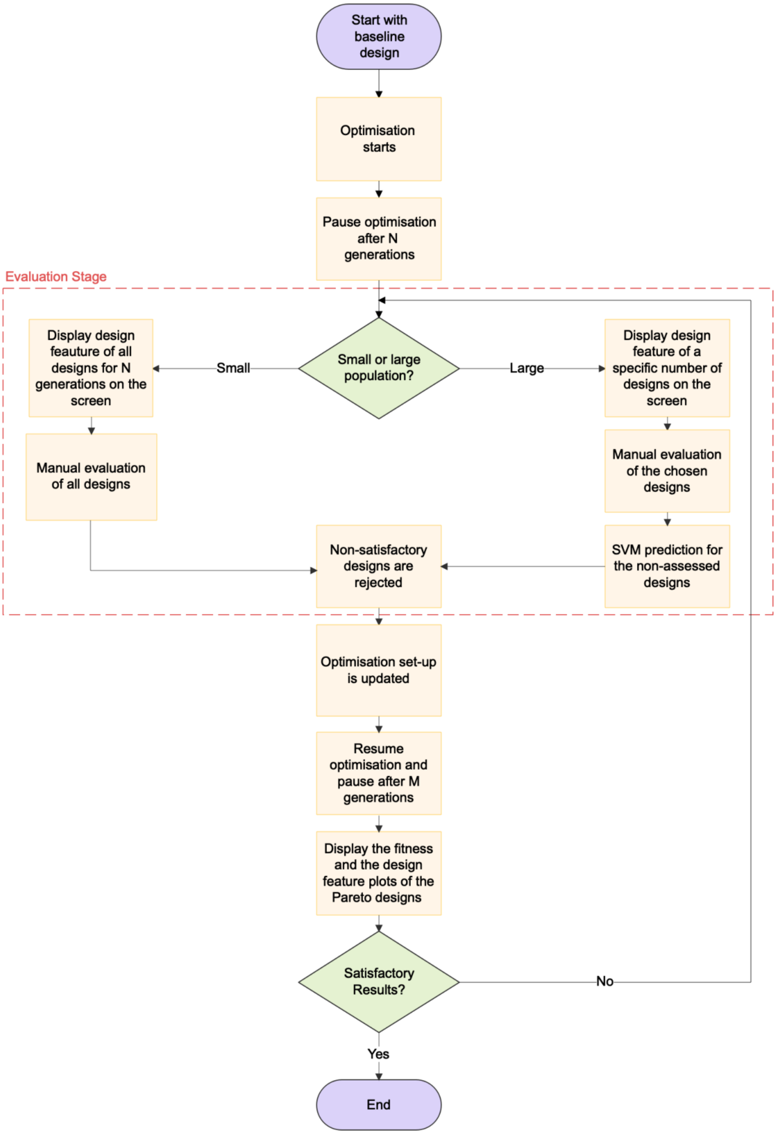


Figure 4.2: *IGA Methodology Flowchart*

4.2 Support-Vector Machines

Support-vector machines are supervised machine learning models, based on Vapnik – Chervonenkis theory [32]. They are used for data analysis for binary classification [4, 33], regression analysis [5] and more recently for multiclass classification [6]. In our methodology, a SVM model is implemented for binary classification of the designs, based on whether they have accepted or rejected design features. It is used in scenarios with large population sizes for reducing the number of user evaluations. More specifically, a smaller subset of the total number of designs is displayed on screen and the users accept or reject the designs. The accepted and rejected designs get the values 1 and 0 respectively; this labelling is input in the model and it is trained. Whenever new designs are evaluated, the SVM model is updated with the new information and the margin of the hyperplane is recalculated. After the training process, when new designs are created from the optimisation algorithm, the SVM model classifies them in one of the two classes.

In figure 4.3 a simple binary classification problem is presented, where the input is a sample from data points with two features x1 and x2, which should be separated in two classes. The model is built with information from the input and based on this, when a new data point enters, it is classified into one of the two classes. The classification is achieved by creating a hyperplane that separates the data in classes by maximising the margin between them. The red straight line of the figure is the best hyperplane in this problem, since it maximises the margin $\frac{2}{\|\beta\|}$. The two parallel lines to the red lines are the marginal hyperplanes and the four points on the hyperplanes are the support-vector and they define the boundary of the margin. The goal when using SVMs for binary classification is to find the optimal hyperplane that classifies the data correctly and to maximise the margin between the classes.

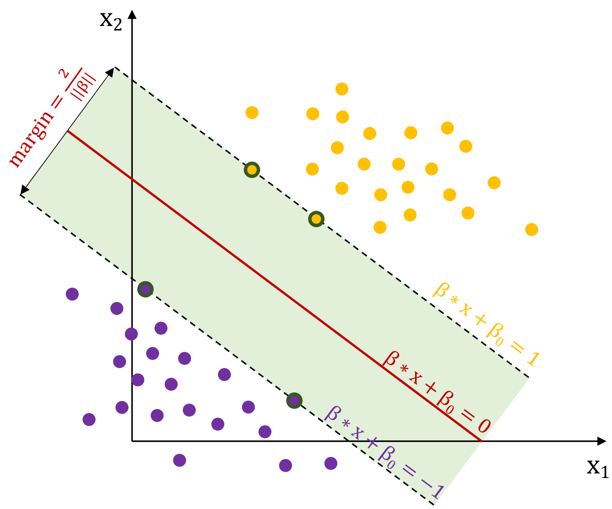


Figure 4.3: *SVM with two classes that are separated from the optimal hyperplane with the largest margin.*

A brief mathematical formulation for the binary classification follows according to [13]. A training set $(x_i,\, y_i)$ is assumed that is comprised of the multidimensional training data x_i, where $i = 1, \dots, n$ and $x_i \in \mathbb{R}^d$, and the class labels $y_i \in \{-1, 1\}$. The hyperplane is defined as,

$$x^T\beta + \beta_0 = 0, \tag{4.1}$$

where β is a unit vector with $\parallel \beta \parallel = 1$ and β_0 a real number.

Additionally, the marginal hyperplanes are defined as $x^T\beta + \beta_0 = 1$ and $x^T\beta + \beta_0 = -1$. Thus the data points should be classified according to,

$$y_i(x_i^T\beta + \beta_0) \geq 1,\ i = 1, \dots, n. \tag{4.2}$$

The optimal hyperplane can be found by solving the convex optimisation problem with linear inequality constraints,

$$\begin{aligned} &\min_{\beta,\beta_0} && \parallel \beta \parallel \\ &\text{subject to} && y_i(x_i^T\beta + \beta_0) \geq 1,\ i = 1, \dots, n. \end{aligned} \tag{4.3}$$

The above description regards data that can be completely linearly separable. However, in real-world applications linear separation is impossible. A solution to this is to use soft margins that are hyperplanes that separate the data by allowing some misclassifications with the introduction of non-negative slack variables $\xi = (\xi_1, \dots, \xi_n)$. In this case, the optimal hyperplane can be calculated by solving the optimisation problem,

$$\begin{aligned} &\min_{\beta,\beta_0} && \frac{1}{2} \parallel \beta \parallel^2 + C\sum_{i=1}^{N} \xi_i \\ &\text{subject to} && \xi_i \geq 0,\ y_i(x_i^T\beta + \beta_0) \geq 1 - \xi_i,\ \forall i\ , \end{aligned} \tag{4.4}$$

where C is a penalty parameter for the compromise between margin maximisation and training error minimisation and $\sum_{i=1}^{N} \xi_i$ is the training error. Lower and higher values of C correspond to models with strict and loose separation respectively.

Finally, in the case of non-linearly separable data, kernels can be used, which map the input data into a high dimensional space and the optimal hyperplane with the maximum margin is calculated in this space where the data can be linearly separable; this process is called Kernel trick. The Kernel function is described as,

$$K(x, x') = \langle h(x), h(x') \rangle\ , \tag{4.5}$$

where $h(x_i)$ are the transformed feature vectors. Broadly used kernels are the linear, the polynomial, the radial basis function and the sigmoid.

5 Results

In this chapter, the results of the two appended papers are presented.

5.1 Summary of Paper A

In Paper A, the first step towards interactive optimisation is introduced for a simple blade design optimisation problem. The designers select specific areas of preference in the Pareto plots, in order to guide the optimisation towards a specific direction.

5.1.1 Description

According to the developed methodology, the optimisation starts by utilising the NSGA-II and after a predefined number of generations, the optimisation is paused and the Pareto plot of the objectives is presented to the users. The users select interesting, according to the objectives, designs and they use them for the next optimisation round; the selected designs form the first generation of the next round. At the same time, the designers have the possibility to change some critical optimisation parameters, the mutation and the crossover operators, with the aim to shift from the exploration of the design space to the exploitation. The search of the design space is now more focused on areas that have been defined from the user-selected designs. The process is repeated until satisfactory results are obtained. The optimisation method is presented in the flowchart of figure 5.1.

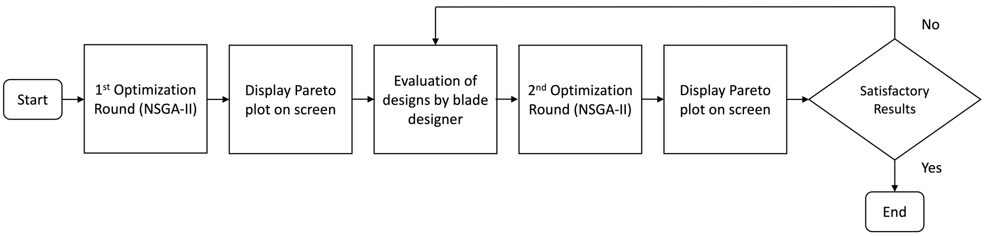


Figure 5.1: *Flowchart of IEC Algorithm for Blade Design Optimisation*

The user scenario regards the design of a controllable pitch propeller for a wind-assisted vessel. There are nine design parameters: the pitch over the propeller diameter, the maximum camber over the chord length of the blade and the skew angle at 20%, 70% and 100% blade radii. Additionally, the objectives are the maximisation of the efficiency in two different operational conditions, at the MCR, where there is the engine's maximum continuous rated power output [1], and at the 50% of MCR. Both conditions have the same speed and there is one constraint related to the geometry of the designs.

5.1.2 Different assessment areas

According to this scenario, the users choose two areas of interest of the Pareto frontier after the end of the first optimisation round. A small area that promotes all designs with efficiency higher than 0.68 in the 50% MCR condition (approach 1). The second area consists of all designs of the first round (approach 2). Figure 5.2 presents the Pareto plot of the first optimisation round, together with the assessment areas of the two approaches. Figures 5.3a and 5.3b show how the optimisation evolved during the second round according to the two approaches. It is evident that by selecting specific areas of interest, the algorithm searches solutions in a more targeted manner. During the second optimisation round, both approaches are examined with 5 different combinations of crossover and mutation operators and each combination is run for 20 times. Figure 5.4 shows the percentage of objective 2 values that are larger than 0.68 along with the confidence interval for both approaches. The first approach reduces the search space more than the second, since the percentage of designs having a 50% MCR efficiency higher than 0.68 is larger than in the second approach. Both approaches have been run for the same number of generations, but the first approach has a smaller population; hence less computational effort is required, compared to the second approach. In addition to this, when the mutation is equal to 0, the percentages are the highest for both approaches in comparison with all other crossover-mutation combinations. More analytical results related to the frequency of the values of the previously described objective, for 5 crossover-mutation combinations, 20 runs and both approaches are presented and discussed in the appended paper.

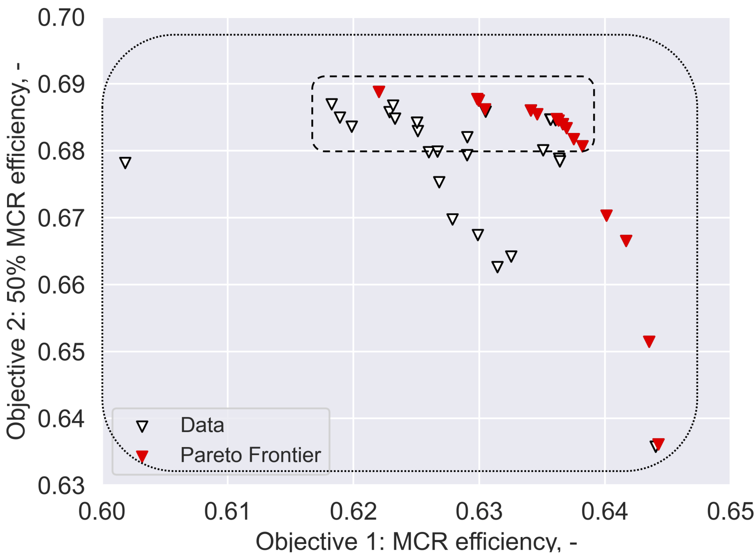
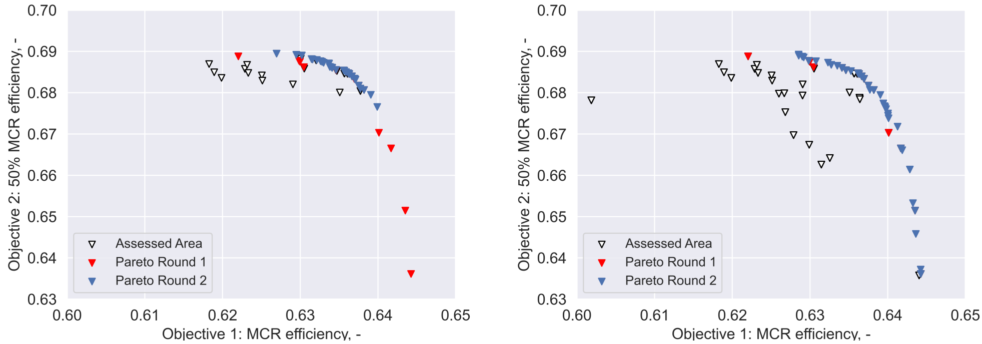


Figure 5.2: *Pareto plot - Optimisation round 1 with assessment areas*



(a) *Approach 1* (b) *Approach 2*

Figure 5.3: *Pareto frontier - Optimisation round 2*

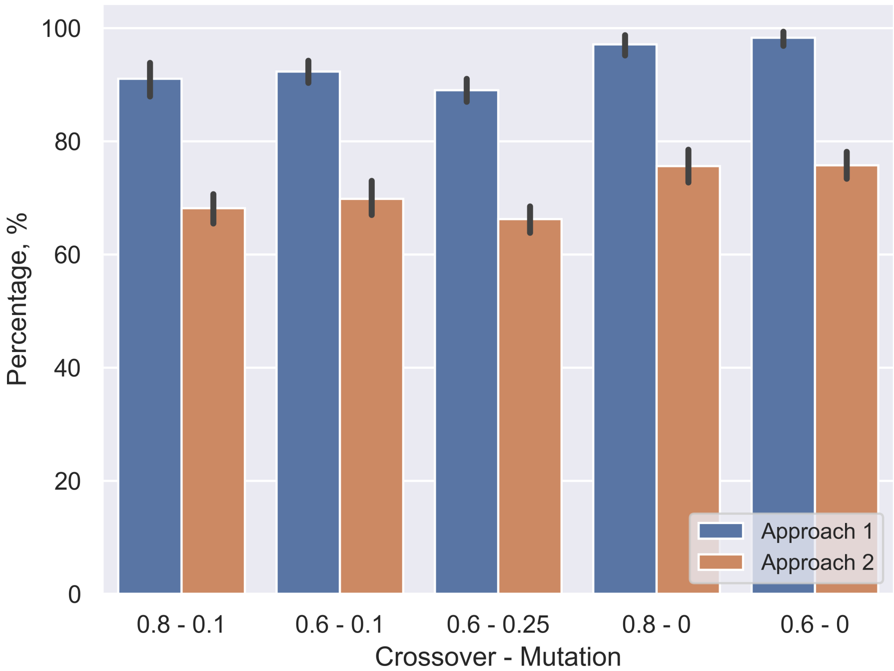
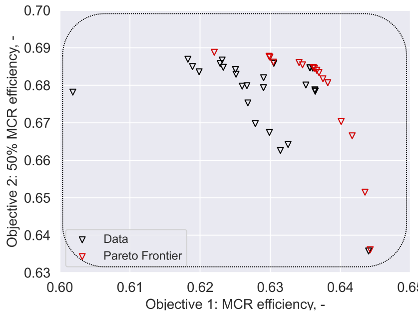


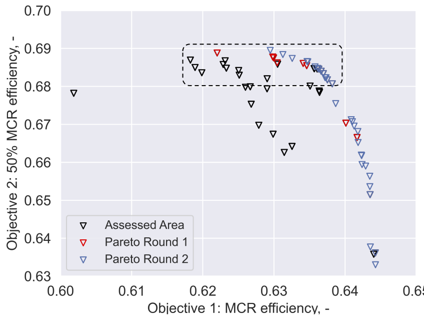
Figure 5.4: *Percentage of second objective values larger than 0.68 with the confidence intervals for both approaches with 5 crossover-mutation combinations*

5.1.3 Interactive optimisation in two steps

The methodology is investigated further by adding one more interactive step in the optimisation process. The designers are called twice to assess two areas of interest and the evolution of the optimisation is presented in figure 5.5. By observing the three Pareto frontiers in 5.5d, the optimisation starts with a coarse frontier. Through the user guidance, a wider frontier is obtained during the second round and finally a detailed frontier that is focused only on the area of interest.

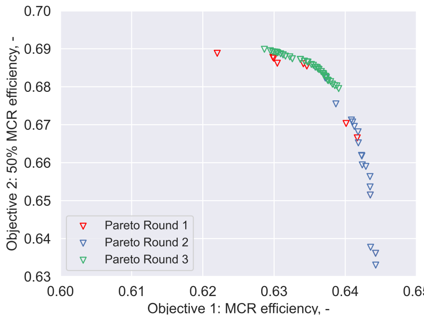


(a) *Round 1:Crossover 0.8 Mutation 0.1*



(b) *Round 2:Crossover 0.8 Mutation 0.1*

(c) *Round 3:Crossover 0.8 Mutation 0*



(d) *The three Pareto frontiers*

Figure 5.5: *Evolution of optimisation algorithm with two interactive steps*

5.1.4 Benefits of the methodology

This methodology is the first step towards interactive optimisation for blade design problems. The results have shown that it is possible to guide the optimisation to a specific direction and the designers obtain a detailed Pareto frontier with several alternatives. Moreover, when the designers select large areas of designs in the Pareto frontiers, it is more beneficial to have a high value in the mutation operator, so that the algorithm searches more broadly solutions in a larger design space, whereas for smaller areas, a lower value in the mutation is preferred.

5.2 Summary of Paper B

In Paper B, the interactive optimisation methodology, as presented in chapter 4, is used for a blade design problem.

5.2.1 Description

The blade designers interact with the optimisation and guide a GA towards the objectives of the design space. The methodology is based on IGAs and the NSGA-II is utilised as the main optimisation algorithm. The blade designers visualise a cavitation related characteristic, evaluate it and this information is input in the optimisation algorithm. SVMs are used as a surrogate model of the methodology for binary classification, in order to solve the user fatigue problem, since blade design requires an optimisation with large populations of individuals and several user evaluations.

The cavitation characteristic that the designer assesses is the cavity that has been developed on the blade at the most critical angle, where there is maximum cavity volume, as shown in figure 5.6. This is a design feature that it is difficult to quantify accurately by using potential methods, thus the user assessment appears to be necessary.

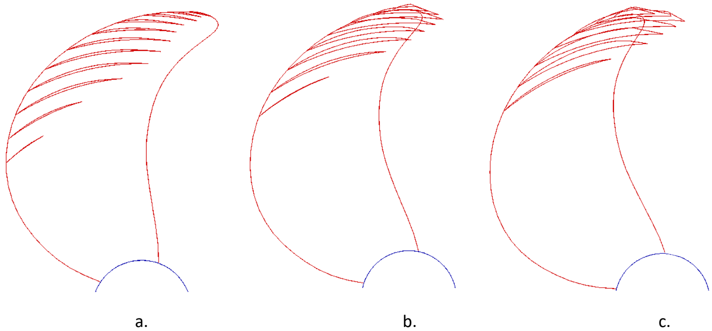


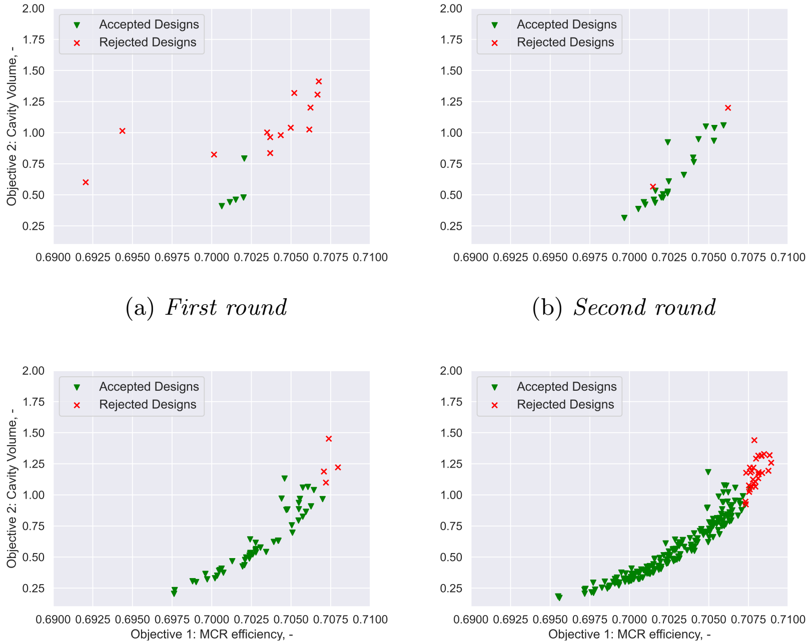
Figure 5.6: *Cavity shape of three propellers at the most critical angle*

The user scenario regards the design of a fixed pitch propeller for a single-screw car-carrier. There are six design parameters: the pitch over the propeller diameter and the maximum camber, at 14%, 70% and 100% (pitch) or 95% (camber) blade radii and range ±15% from the values of the baseline design. There are two objectives: the maximisation of the efficiency and the minimisation of the maximum cavity volume at the MCR condition. Finally, there is one quantitative constraint, a pitch adjustment based on given threshold values for propeller thrust (±2% from the thrust value of the baseline design).

In order to investigate the capabilities of the IGA methodology thoroughly, the interactive methodology is performed with and without the SVM model and it is compared with automated optimisation as well. The results are summarised in the following three sections.

5.2.2 Interactive optimisation

For the interactive optimisation four sequential optimisation rounds are run with a total of 1140 designs, where the users are asked to interact with the code in between these rounds three times. The first three rounds are small, in order to avoid user fatigue from many evaluations. This is shown in figure 5.7, where the plots of the two objectives for all four optimisation rounds are presented and the designs are divided in two groups of accepted and rejected with green and red colour respectively. The optimisation started with a small population of mostly rejected designs (figure 5.7a) and finished with a large population of designs, where most of them are accepted (figure 5.7d).



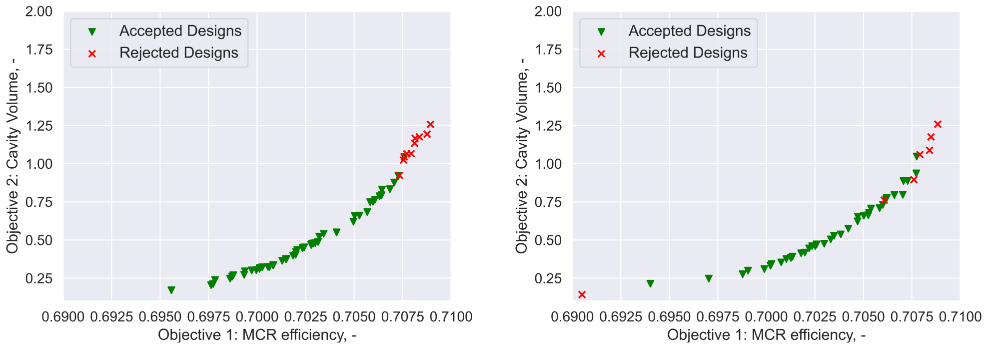
(a) *First round* (b) *Second round*

(c) *Third round* (d) *Fourth round*

Figure 5.7: *Accepted and rejected designs of four optimisation rounds*

5.2.3 Comparison between interactive and automated approaches

For performing the automated optimisation, the same user scenario was used for a total of 1100 designs, in order to be able to make a comparison with the 1140 designs of the interactive optimisation. The user pre-defines the design and optimisation parameters, the optimisation runs and they obtain a Pareto frontier with all non-dominated designs in the end. By comparing the Pareto frontiers of the two approaches in figures 5.8a and 5.8b, the interactive optimisation gives 49 accepted and 10 rejected designs and the automated optimisation gives 38 and 7 respectively. This means that the user obtains 11 more design alternatives with the interactive optimisation.

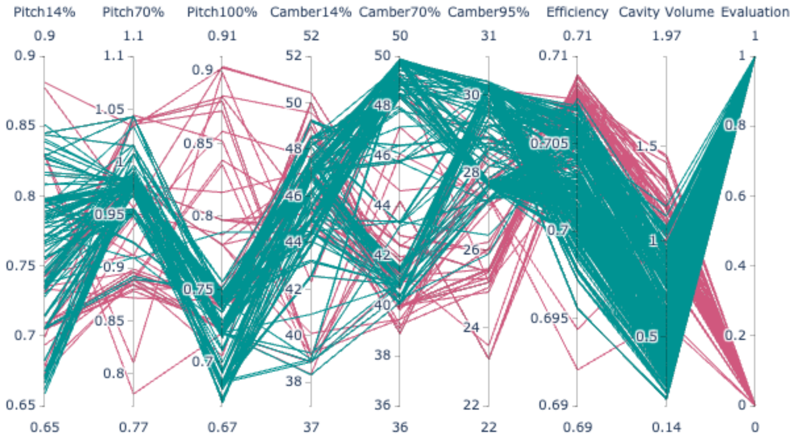


(a) *Interactive optimisation*

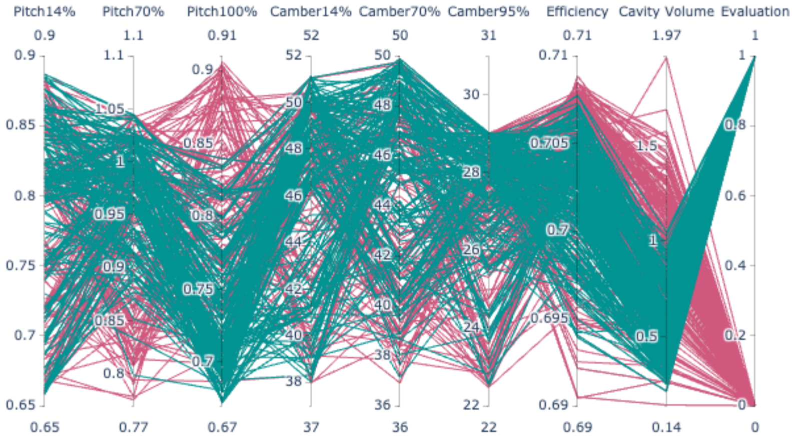
(b) *Automated Optimisation*

Figure 5.8: *Final Pareto frontiers of two approaches*

Another important part of the optimisation analysis is how the design space has been explored. The automated algorithm generates 436 unique designs and the interactive 236 unique designs. Figure 5.9 presents where these designs have been generated. It shows the parallel coordinates' visualisation of the design variables, objectives and user evaluation for both approaches. Each line represents one design and shows the values of its design variables, objectives and user evaluations. The green and red colours of the lines depict the accepted and rejected designs respectively. It is apparent that the design variables of the interactive approach have not been explored as broadly as in the automated one. However, the search of the interactive optimisation is more targeted on areas where there are accepted designs. This is evident in figure 5.10, where only the accepted designs are shown. For the automated approach, the algorithm has not searched in areas, where the camber at 95% radius is as increased as in the interactive approach. 26 individuals, where the camber at 95% radius has values higher than 30 mm ended up being in the final Pareto frontier of the interactive optimisation.

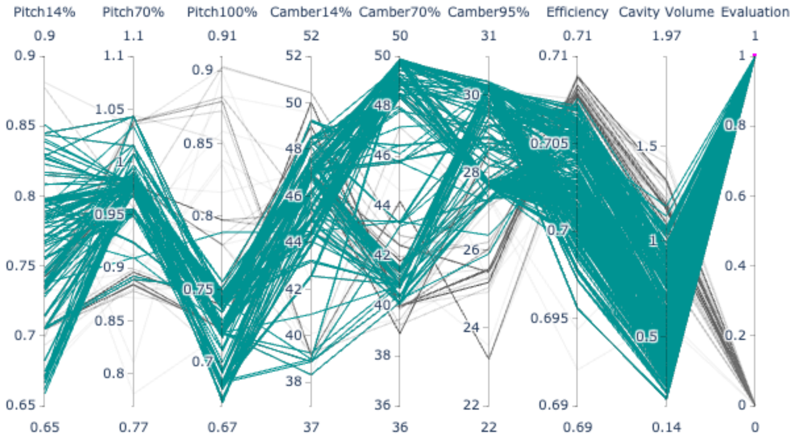


(a) *Interactive optimisation*

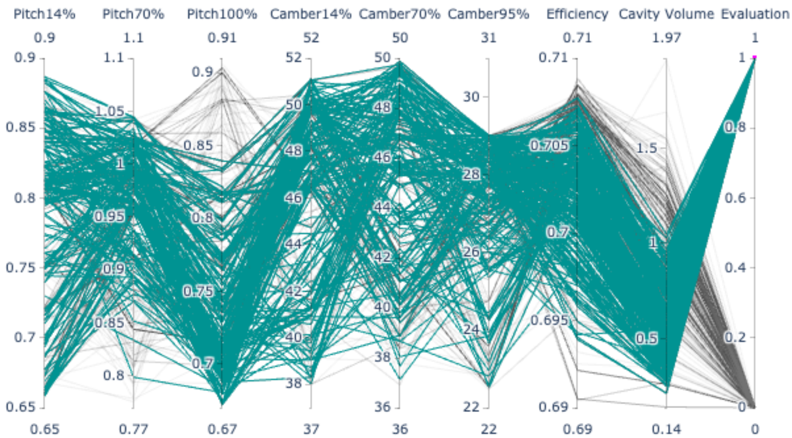


(b) *Automated Optimisation*

Figure 5.9: *Parallel coordinate visualisation of design variables, objectives and evaluation*



(a) *Interactive optimisation*



(b) *Automated Optimisation*

Figure 5.10: *Parallel coordinate visualisation - accepted designs*

5.2.4 Interactive Optimisation with SVM implementation

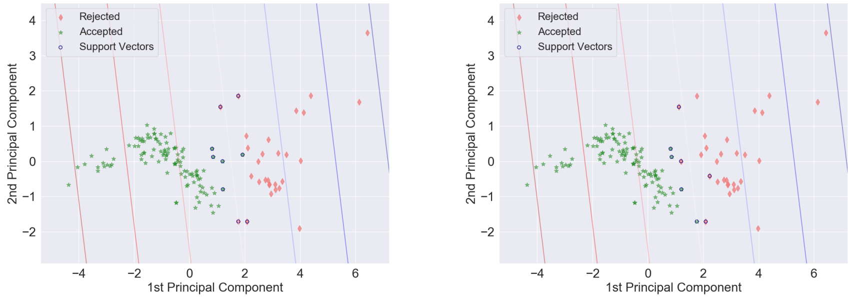
The results of the IGA methodology were presented in 5.2.2 without including the SVM model. After the end of the fourth optimisation round, all designs were manually evaluated by the user, with the purpose of showing precisely how the code evolves according to the user preference. However, this is a labour-intensive process for the designers and can be improved by using the SVM model during the last stage of the optimisation. The user manually evaluated 99 designs in the first three optimisation rounds by classifying them in groups of accepted or rejected designs and this can be used as the training input of the SVM model. The input trains the SVM model and when the final round finishes, the model classifies the new designs as accepted or rejected. The total number of unique designs were 236, so the train data are 42% of the total and the test data the remaining 58%. For the validation of the accuracy of the SVM model, the model's predicted classification was compared to the manual evaluations of section 5.2.2. As shown in table 5.1, by using either the linear or the polynomial Kernel the predictability is high, 98.5% and 97% respectively.

Table 5.1: SVM predictability of last optimisation round in interactive approach

% of training data	Predictability	
	Linear Kernel	Polynomial Kernel
42	0.985	0.97

Figure 5.11 presents the class separation of the designs, by using a linear kernel, with the aid of the principal component analysis (PCA) [16]. The data from the manual evaluations are presented in figure 5.11a. The penalty parameter C is set equal to 1 and there are two misclassified points (outliers) from the separating hyperplane. In figure 5.11b, the predicted values are presented that have four outliers. It is evident that the predictability of the model is satisfactory.

Additionally, the data of the automated approach were used to do a sensitivity analysis on the SVM model's prediction accuracy by comparing the SVM prediction with the manual evaluations. This is shown in the summary statistics of figure 5.12 for the linear kernel. Different training data were chosen randomly as subsets of the whole data set and were input in the SVM; 9 different sizes of training data were considered and the process was repeated 150 times for each size. It is observed that when the training data is 50-90% of the data set the mean accuracy is approximately the same, with the lower standard deviation being at 50-70% of the data set. According to these results, for the specific problem, the user should choose to manually evaluate 50% of the unique designs, in order to have as few manual evaluations as possible with accurate prediction.



(a) *Manual Evaluations* (b) *Predicted Data*

Figure 5.11: *Class separation in SVM model with PCA - Linear Kernel*

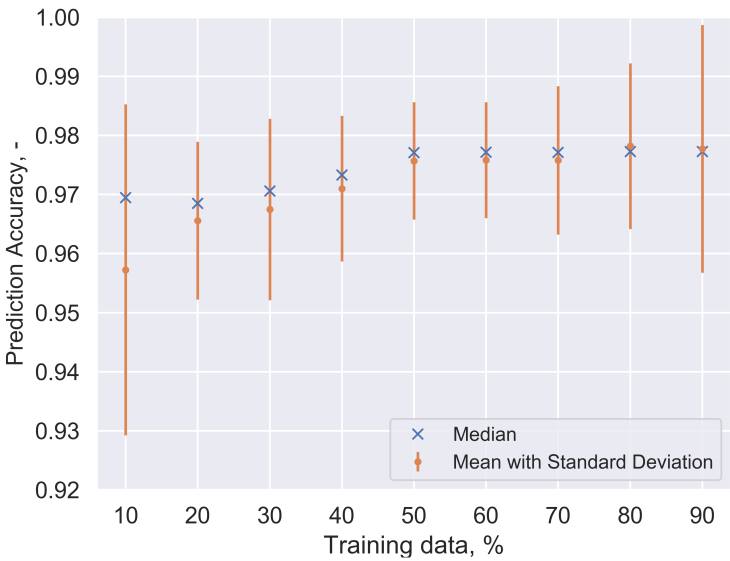


Figure 5.12: *Summary statistics of SVM prediction accuracy - Linear Kernel*

5.2.5 Benefits of the methodology

Through the proposed methodology, the results show that the user obtains at the end of the optimisation a detailed Pareto frontier, with non-dominated and feasible design solutions that have a good cavitation behaviour in line with the designer's preference. By doing the user evaluations in different steps, with the optimisation running in between them, less fatigue is caused to the users. When compared to the results of the automated approach, both have approximately the same performance, but in the automated approach the design space is searched more broadly; at the same time most of those diverse solutions are rejected by the users, as they do not fulfil the cavitation requirements. Thus the interactive approach finds more solutions in a more targeted manner without performing unnecessary calculations. Regarding the SVM model, it is needed when the populations are large in order to reduce user evaluations and its predictability accuracy proved to be satisfactory.

6 Concluding Remarks

Marine propeller design is a multidisciplinary and multi-objective process that involves contradicting requirements from the stakeholders and necessitates several iterations in between its stages in order to fulfil those requirements. Although creating designs manually results in good propeller geometries, it has proved to be labour-intensive, since there are multiple design parameters, resulting in a large design space. In recent years, automated optimisation has been used in some blade design applications, which has the benefit of producing numerous design alternatives fast. If the algorithm converges in good manner, which very often is not certain, the designs show increased performance. The blade designers obtain at the end of any optimisation process, a Pareto frontier with all the optimal non-dominated solutions and they select the design that represents the best trade-off according to the requirements. It is very common though that the Pareto frontier consists of solutions that represent infeasible propeller geometries. This happens because with the available fast tools of the optimisation, important phenomena are not taken into consideration and they cannot be quantified accurately, in order to be included as constraints or objectives in the optimisation. Therefore, a need has appeared for enabling the designers to have more control over the whole design and optimisation procedure. This can be done by creating a process, where the blade designers interact with the optimisation algorithms throughout the optimisation and their elicited information is later taken into account for the continuation of the optimisation.

The development of such an interactive optimisation process is presented in this thesis. The main purpose through interactive optimisation is to assist and support the blade designers during the design process and enable them to interact with the design tools during the optimisation. In Paper A, an interactive optimisation process was introduced for a blade design problem. We followed a progressively interactive approach, where the users were enabled to select areas of the Pareto frontier and promote them to the following generations. The areas were selected according to the user preference for the specific problem. In Paper B, the methodology was developed further by adding one more interactive step and implementing a surrogate model. More specifically, images that present the cavity on the blade of the designs, are presented on the screen and the users decide whether the developed cavity is acceptable or not. SVMs are used as surrogate model to the optimisation for scenarios that require large populations of individuals.

The main advantage of the proposed methodology is that the designers have control over the optimisation procedure and the evolution of the algorithms. In both papers, the users managed to guide the algorithm towards areas of the design space with solutions that represented designs with the desired performance or other design characteristics. Moreover, through the interactive optimisation, the designers gained very detailed Pareto frontiers that offer several design alternatives. In addition to this, in Paper B, the Pareto frontier of the interactive optimisation had more accepted designs in terms of cavitation, when compared to the automated optimisation. This means that the interactive approach finds more solutions in a more targeted manner without performing unnecessary calculations. Another important finding for Paper A, is that the selection of the mutation operator plays an important role in the search of the design space. It appears that by initiating the optimisation with larger values of mutation leads to search in a wider design space, while decreasing the mutation narrows down the search of the space. Thus, depending on whether the users select large or small areas of designs in the Pareto frontier, the values of the mutation operator should be higher or lower respectively. In Paper B, by implementing the SVM model in the methodology, the optimisation procedure was accelerated and the user fatigue burden was diminished. The model's prediction accuracy proved to be satisfactory, by training just a small sample (42%) of the total dataset. Also, by performing user evaluations in different steps gradually, with the optimisation running in between them, less fatigue is caused to the users.

The proposed methodology is the beginning stage for a more systematic user-code interaction in blade design optimisation problems. The benefits of the methodology will be more evident in more complex scenarios though, which involve more objectives and constraints. In the current scenario, only one operational condition was considered, but it is common practice in blade design to optimise propellers by taking into consideration more conditions. Especially with the increase of wind-powered and wind-assisted vessels nowadays, where more design points and operational conditions have to be considered, the use of interactivity will be important. Through the proposed methodology, the design space will not be constrained from the beginning of the optimisation, but gradually through the information from the user assessments. It is also important to extend the methodology in order to include the variation of the propeller load due to the use of the wind and other aspects that are connected to wind-assisted propulsion.

An essential part of interactive optimisation is how the interactivity takes place and it is necessary to focus more on the further development of the graphical user interface in the proposed method. A human factor research with the participation of blade designers could aid towards that direction, in order to define how many designs should be presented simultaneously, if there should be a reference design for comparison, if the users believe that they should be enabled to alternate the geometry of the designs, after how many user evaluations do they feel fatigued etc. This process will also create ideas on more new interactive steps that could lead to faster convergence and to more efficient solutions.

In Paper B, SVMs were the only machine learning method that was utilised as a surrogate model. More methods will be used, in order to investigate their potential for improving the performance of the interactive optimisation. In addition to this, in the proposed methodology, the training sample for the machine learning model is selected randomly out of the total unique designs of the optimisation. However, it is of interest to look into more effective ways of selecting the training sample of the machine learning model, with the aim to further eliminate unnecessary user evaluations.

References

[1] J. Babicz. *Wärtsilä Encyclopedia of Ship Technology*. Wärtsilä Corporation, 2015. ISBN: 9789529355365. URL: https://books.google.se/books?id=4BtNswEACAAJ.

[2] S. Berger, M. Druckenbrod, M. Pergande, and M. Abdel-Maksoud. A Two-Stage Optimisation Method for Full-Scale Marine Propellers Working Behind a Ship. *Ship Technology Research* **61**.2 (2014), 64–79. DOI: 10.1179/str.2014.61.2.001. URL: https://doi.org/10.1179/str.2014.61.2.001.

[3] J. Biles, P. Anderson, and L. Loggi. "Neural network fitness functions for a musical IGA". 1996.

[4] C. Cortes and V. Vapnik. Support Vector Networks. *Machine Learning* **20** (1995), 273–297.

[5] H. Drucker, Chris, B. L. Kaufman, A. Smola, and V. Vapnik. "Support vector regression machines". *Advances in Neural Information Processing Systems 9*. Vol. 9. 1997, pp. 155–161. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.21.5909.

[6] K. bo Duan and S. S. Keerthi. "Which is the best multiclass SVM method? An empirical study". *Proceedings of the Sixth International Workshop on Multiple Classifier Systems*. 2005, pp. 278–285.

[7] J. Felkner, E. Chatzi, and T. Kotnik. "Interactive particle swarm optimization for the architectural design of truss structures". *2013 IEEE Symposium on Computational Intelligence for Engineering Solutions (CIES)*. 2013, pp. 15–22. DOI: 10.1109/CIES.2013.6611723.

[8] B. Filipič and D. Juričić. "An Interactive Genetic Algorithm for Controller Parameter Optimization". *Artificial Neural Nets and Genetic Algorithms*. Ed. by R. F. Albrecht, C. R. Reeves, and N. C. Steele. Vienna: Springer Vienna, 1993, pp. 458–462. ISBN: 978-3-7091-7533-0.

[9] S. Gaggero, G. Tani, D. Villa, M. Viviani, P. Ausonio, P. Travi, G. Bizzarri, and F. Serra. Efficient and multi-objective cavitating propeller optimization: An application to a high-speed craft. *Applied Ocean Research* **64** (Mar. 2017), 31–57. DOI: 10.1016/j.apor.2017.01.018.

[10] F. Goulart and F. Campelo. Preference-guided evolutionary algorithms for many-objective optimization. *Information Sciences* **329** (2016), 236–255.

[11] I. Gypa, R. Bensow, K. Wolff, and R. Gustafsson. "Interactive Evolutionary Computation for Propeller Design Optimization of Wind-Assisted Vessels". *AIAA AVIATION 2020 FORUM*. 2020. DOI: 10.2514/6.2020-3162.

[12] I. Gypa, M. Jansson, K. Wolff, and R. Bensow. "Propeller optimisation by interactive genetic algorithms and machine learning". Manuscript submitted for publication in Ship Technology Research journal.

[13] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., 2001.

[14] K. O. Holden, O. Fagerjord, and R. Frostad. *Early design-stage approach to reducing hull surface forces due to propeller cavitation*. Tech. rep. 1980.

[15] J. Huisman and E. Foeth. "Automated multi-objective optimization of ship propellers". 2017.
[16] I. Jolliffe. *Principal Component Analysis*. Springer-Verlag New York, 2002. DOI: 10.1007/b98835.
[17] Kongsberg hydrodynamic design team. Private Communication. Kristinehamn, Sweden, 2020.
[18] J.-H. Lee and M.-L. Chang. Stimulating designers' creativity based on a creative evolutionary system and collective intelligence in product design. *International Journal of Industrial Ergonomics* **40**.3 (2010), 295–305.
[19] A. Lewis and T. Kipouros. "Methods of Interactive Optimisation in Engineering Design". Feb. 2015. DOI: 10.13140/2.1.1706.8320.
[20] X. Llorà, K. Sastry, D. Goldberg, A. Gupta, and L. Lakshmi. "Combating user fatigue in iGAs: partial ordering, support vector machines, and synthetic fitness". *GECCO '05*. 2005.
[21] J. Madar, J. Abonyi, and F. Szeifert. Interactive evolutionary computation in process engineering. *Computers & chemical engineering* **29**.7 (2005), 1591–1597.
[22] M. Ohsaki and H. Takagi. Improvement of presenting interface by predicting the evaluation order to reduce the burden of human interactive EC operators. *SMC'98 Conference Proceedings. 1998 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No.98CH36218)* **2** (1998), 1284–1289 vol.2.
[23] E. Poirson, J.-F. Petiot, L. Boivin, and D. Blumenthal. Eliciting user perceptions using assessment tests based on an interactive genetic algorithm. *Journal of Mechanical Design* **135**.3 (2013).
[24] Y. Ren and P. Y. Papalambros. A design preference elicitation query as an optimization process. *Journal of Mechanical Design* **133**.11 (2011).
[25] C. Simons, J. Smith, and P. White. Interactive ant colony optimization (iACO) for early lifecycle software design. *Swarm Intelligence* **8** (2 2014), 139–157. ISSN: 1935-3812. DOI: 10.1007/s11721-014-0094-2. URL: https://uwe-repository.worktribe.com/output/817012.
[26] T. W. Simpson and J. R. Martins. Multidisciplinary design optimization for complex engineered systems: report from a national science foundation workshop. *Journal of Mechanical Design* **133**.10 (2011).
[27] A. Sinha, P. Korhonen, J. Wallenius, and K. Deb. An interactive evolutionary multi-objective optimization algorithm with a limited number of decision maker calls. *European Journal of Operational Research* **233**.3 (2014), 674–688.
[28] X. Sun, D. Gong, and W. Zhang. Interactive genetic algorithms with large population and semi-supervised learning. *Appl. Soft Comput.* **12** (2012), 3004–3013.
[29] H. Takagi. Interactive evolutionary computation: fusion of the capabilities of EC optimization and human evaluation. *Proceedings of the IEEE* **89**.9 (2001), 1275–1296.
[30] J. Tornblad and K. A. M. Laboratory. *Marine Propellers and Propulsion of Ships*. Marine Laboratory, KaMeWa, 1987.
[31] F. Valdenazzi, F. Conti, S. Gaggero, D. Grassi, C. Vaccaro, and D. Villa. "A Practical Tool for the Hydro-Acoustic Optimization of Naval Propellers". May 2019.

[32] V. Vapnik and A. Chervonenkis. *Theory of Pattern Recognition [in Russian]*. Nauka, 1974.
[33] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Berlin, Heidelberg: Springer-Verlag, 1995. ISBN: 0387945598.
[34] F. Vesting. "Marine Propeller Optimisation - Strategy and Algorithm Development." PhD thesis. Chalmers University of Technology, 2015.
[35] F. Vesting, R. Bensow, R. Johansson, R. Gustafsson, and N. A. Costa. Procedure for Application-Oriented Optimisation of Marine Propellers. *Journal of Marine Science and Engineering* **4** (2016).
[36] F. Vesting, R. Gustafsson, and R. Bensow. Development and application of optimisation algorithms for propeller design. *Ship Technology Research* **63** (2016), 50–69.
[37] M. Wahde. *Biologically inspired optimization methods: an introduction*. WIT press, 2008.
[38] B. Yannou, M. Dihlmann, and R. Awedikian. "Evolutive Design of Car Silhouettes". *DAC 2008*. 2008.